Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 9 (2012) 286 – 291

Procedia Computer Science

International Conference on Computational Science, ICCS 2012

Physically accurate granular flow simulation

S. Oliveira[a], D.E. Stewart[b,]

[a]*Department of Computer Science, University of Iowa, Iowa City, IA 52242*
[b]*Department of Mathematics, University of Iowa, Iowa City, IA 52242*

Abstract

Discrete element simulation models large collections of solid bodies as individual bodies, rather than attempt to approximate the system via continuum models and partial differential equations. However, standard models for rigid bodies are inadequate as they do not incorporate elastic vibrations. An alternative to both rigid body models and finite element simulations is presented that is highly parallelizable. These models are the basis for GPU implementations of discrete element simulation.

Keywords: discrete element simulation, elastic bodies, parallel simulation

1. Introduction

Discrete element simulation is used to model flow and movement of granular materials, such as gravel, grains, sand and powders. Important applications of granular flow include rock slides, size sorting by shaking, and various manufacturing processes involving many parts such as boxing nails or screws. Solid bodies are typically modeled as rigid bodies , which is generally a good approximation to reality. However, impact models for rigid bodies [1, 2] are problematic as they require a "coefficient of restitution" which measures how elastically a body bounces.

While it is natural to assume that this coefficient of restitution is a material property (since rubber balls bounce well, but soft clay balls do not), this is not in fact the case. For example, the experiments of Stoianivici & Hurmuzlu [3] with thick rods of different aspect ratios showed a very strong dependence of the kinematic coefficient of restitution on the *angle* of impact. Further difficulties were found by [4, 5] with various impact models. The resolution of the difficulties lies not with more sophisticated impact models based on rigid bodies. Rather the fundamental issue is that elasticity must be incorporated into the model. One-dimensional models of rods going back to Routh [6] (1860) show that elastic rods in impact have elastic waves traversing the rod which explains their coefficient of restitution.

Incorporating elasticity into the models of impact typically involve partial differential equations, which are computationally very expensive in both time and memory, especially when we are dealing with thousands of bodies in contact or collision. In addition, because of the tight coupling between nodes in a finite element model, it is harder to parallelize these methods. A further computational difficulty is the Courant–Friedrichs–Levy (CFL) limit on the time step used for solving the equations of elasticity [7]. This essentially limits the time step to be no more than the time an elastic wave takes to traverse the smallest element of the finite element model. Even though this limit is a necessary condition only for explicit methods for solving the differential equations in time, if implicit methods are used, violating the CFL limit by a wide margin results in poor numerical simulations.

In this paper we propose a new approach based on convolution complementarity problems [8] which is highly parallelizable, incorporates elastic waves and vibrations, and is less computationally intensive that full finite element modeling. It is thus a much better way of simulating large numbers of contacting and impacting bodies.

doi:10.1016/j.procs.2012.04.030

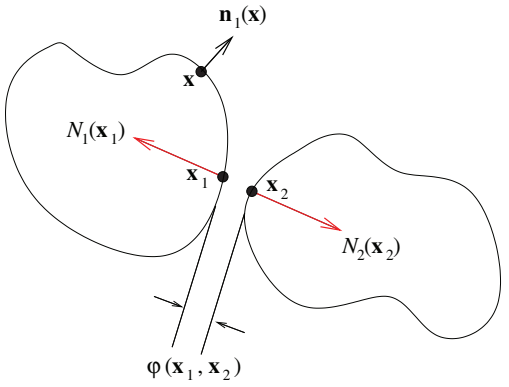

Figure 1: Geometry of bodies near contact

2. Convolution complementarity models of impact

2.1. Contact conditions

For frictionless contact between rigid bodies, there is a natural complementarity relationship between the contact force acting between them [1]:

$$0 \le \text{separation} \perp \text{contact force} \ge 0.$$

That is, both the separation and the contact force are non-negative, the (dot) product of the two is zero. In other words, both the separation and contact force are non-negative, and if one is positive the other is zero.

For elastic bodies, there are infinitely many points of possible contact, and the contact force can act at each and every point of contact. At each point of potential contact, this complementarity relationship must hold. To formalize this relationship we introduce some notation. Let $\mathbf{u}(t, \mathbf{x})$ be the displacement or deformation of the point $\mathbf{x}$ of an elastic body $\Omega \subset \mathbb{R}^3$ at time t. At every point $\mathbf{x}$ on the boundary $\partial\Omega$ of Ω we have the outward normal vector $\mathbf{n}(\mathbf{x})$. Assuming small deformations, the separation between between $\mathbf{x}$ and a rigid obstacle is given by

$$\text{separation} = \varphi(\mathbf{x}) - \mathbf{n}(\mathbf{x}) \cdot \mathbf{u}(t, \mathbf{x})$$

where $\varphi(\mathbf{x})$ is the separation of the undeformed body from the obstacle.

For a pair of elastic bodies, Ω_1 and Ω_2, a similar formula can be used to determine the separation between the bodies at points $\mathbf{x}_1 \in \partial\Omega_1$ close to $\partial\Omega_1$. Fix a mapping $\mathbf{x}_1 \mapsto \pi_{1,2}(\mathbf{x}_1) = \mathbf{x}_2 \in \partial\Omega_2$; we could take $\mathbf{x}_2$ to be the nearest point in $\partial\Omega_2$, but need not be. Then the separation between $\mathbf{x}_1$ and $\mathbf{x}_2$ is given approximately for $\mathbf{x}_1 \approx \mathbf{x}_2$ by:

$$\begin{aligned} &\text{separation} \\ &\quad \approx \varphi(\mathbf{x}_1, \mathbf{x}_2) - \mathbf{n}_1(\mathbf{x}_1) \cdot [\mathbf{u}_1(t, \mathbf{x}_1) - \mathbf{u}_2(t, \mathbf{x}_2)] \\ &\quad \approx \varphi(\mathbf{x}_1, \mathbf{x}_2) - \mathbf{n}_2(\mathbf{x}_2) \cdot [\mathbf{u}_2(t, \mathbf{x}_1) - \mathbf{u}_1(t, \mathbf{x}_2)] \end{aligned}$$

since $\mathbf{n}_1(\mathbf{x}_1) \approx -\mathbf{n}_2(\mathbf{x}_2)$. Here $\varphi(\mathbf{x}_1, \mathbf{x}_2)$ is the separation between $\mathbf{x}_1$ and $\mathbf{x}_2$ with zero displacement or deformation. Ignoring the geometric nonlinearities if $\mathbf{x}_1$ and $\mathbf{x}_2$ are close, we can take these approximations to be equalities, and $\mathbf{n}_1(\mathbf{x}_1) = -\mathbf{n}_2(\mathbf{x}_2)$. Then the normal contact forces on Ω_1 at $\mathbf{x}_1$ is $-N_1(\mathbf{x}_1)\,\mathbf{n}_1(\mathbf{x}_1)$, and similarly on Ω_2 at $\mathbf{x}_2$ is $-N_2(\mathbf{x}_2)\,\mathbf{n}_2(\mathbf{x}_2)$; by Newton's third law of motion, these are negatives of each other, so we take $N_1(\mathbf{x}_1) = N_2(\mathbf{x}_2)$. These relationships are illustrated in Figure 1.

The complementarity conditions that we want to solve for are then

$$\begin{aligned} 0 \le \varphi(\mathbf{x}_1, \mathbf{x}_2) - \mathbf{n}_1(\mathbf{x}_1) \cdot [\mathbf{u}_1(t, \mathbf{x}_1) - \mathbf{u}_2(t, \mathbf{x}_2)] \\ \perp N_1(t, \mathbf{x}_1) = N_2(t, \mathbf{x}_2) \ge 0 \end{aligned} \tag{1}$$

for all $\mathbf{x}_1 \in \partial\Omega_1$ in potential contact with Ω_2, and $\mathbf{x}_2 = \pi_{1,2}(\mathbf{x}_1)$.

2.2. Equations of elasticity

It should be understood that in each body, both the displacement field $\mathbf{u}_i(t,\mathbf{x})$ and the contact forces $N_i(t,\mathbf{x})$ are unknowns to be determined. Connecting them are the equations of elasticity. For frictionless linear elasticity these can be given for body i as:

$$\rho_i(\mathbf{x})\frac{\partial^2 \mathbf{u}_i}{\partial t^2} = \operatorname{div}\sigma_i[\mathbf{u}_i] + \mathbf{f}_i(t,\mathbf{x}), \tag{2}$$

$$\sigma_i[\mathbf{u}_i](t,\mathbf{x})\cdot\mathbf{n}_i(\mathbf{x}) = -N_i(t,\mathbf{x})\,\mathbf{n}_i(\mathbf{x}). \tag{3}$$

Equation (2) holds inside Ω_i, while (3) holds on the boundary $\partial\Omega_i$. The quantity $\mathbf{f}_i(t,\mathbf{x})$ is the external or non-elastic forces acting on body Ω_i at time t and at position $\mathbf{x}\in\Omega_i$. The quantity $\sigma_i[\mathbf{u}_i](t,\mathbf{x})$ is the stress tensor, which is a symmetric rank two tensor or matrix. It can be determined from the linearized strain tensor

$$\varepsilon[\mathbf{u}] = \frac{1}{2}\left(\nabla\mathbf{u} + \nabla\mathbf{u}^T\right)$$

for linear elasticity, by means of the constitutive relations

$$\sigma_i[\mathbf{u}]_{jk} = \sum_{p,q} a^{(i)}_{jkpq}\varepsilon_{pq}[\mathbf{u}]$$

where $a^{(i)}_{kjpq}$ are the material parameters for body Ω_i.

For this paper, the important point is that the equations of elasticity are linear, and independent of time. This means that the solution depends in a linear way on the external forces ($\mathbf{f}_i(t,\mathbf{x})$), initial displacement ($\mathbf{u}_i(0,\mathbf{x})$) and velocity ($\partial\mathbf{u}_i/\partial t(0,\mathbf{x})$), and contact forces ($N_i(t,\mathbf{x})$). Furthermore, because the equations are independent of time, we can write the solution in terms of a convolution in time:

$$\mathbf{u}_i(t,\mathbf{x}) = \mathbf{u}_i^{(0)}(t,\mathbf{x}) + \int_0^t\int_{\partial\Omega_i}\mathbf{w}_i(t-\tau,\mathbf{x};\mathbf{y})\,N_i(\tau,\mathbf{y})\,dS(\mathbf{y})\,d\tau. \tag{4}$$

Here $\mathbf{u}_i^{(0)}(t,\mathbf{x})$ is the displacement field assuming zero contact forces N_i.

2.3. Convolution complementarity problems

We now need to combine the representation of the displacement field solutions (4) with the complementarity condition (1). This gives the complementarity conditions for N_1:

$$0 \le N_1(t,\mathbf{x}) \perp \varphi^{(0)}(t,\mathbf{x}) + \int_0^t\int_{\partial\Omega_1}\mathbf{n}_1(\mathbf{x})\cdot\mathbf{w}_1(t-\tau,\mathbf{x};\mathbf{y})\,N_1(\tau,\mathbf{y})\,dS(\mathbf{y})\,d\tau \ge 0$$

where $\varphi^{(0)}(t,\mathbf{x})$ is the separation between $\mathbf{x}\in\partial\Omega_1$ and Ω_2 that would have occurred if the contact forces were zero. This is still a complementarity condition that involves infinitely many points $\mathbf{x}_1\in\partial\Omega_1$. To create a computationally feasible problem, we represent

$$N_1(t,\mathbf{x}) = \sum_{j=1}^m \psi_{1,j}(\mathbf{x})\,N_{1,j}(t)$$

where $\psi_{1,j}$ are smooth non-negative functions on $\partial\Omega_1$, and require complementarity between $N_{1,j}(t)$ and $\int_{\partial\Omega_1}\psi_{1,j}(\mathbf{x})\,\text{separation}(t,\mathbf{x})\,dS(\mathbf{x})$. This gives a finite dimensional convolution complementarity problem (CCP) [8] of the following form:

$$0 \le N_{1,j}(t) \quad\perp\quad \varphi^{(0)}_{1,j}(t) + \int_0^t\sum_{k=1}^m \mu_{jk}(t-\tau)\,N_{1,k}(\tau)\,d\tau \ge 0,$$

$j=1,2,\ldots,m$. The kernel function of this convolution $\mu(t)=\left[\mu_{jk}(t)\right]$ is a $m\times m$ matrix-valued function that captures the elastic behavior of the body. In matrix-vector form, it can be written as

$$0 \le \boldsymbol{N}_1(t) \perp \boldsymbol{\varphi}^{(0)}_1(t) + \int_0^t \mu(t-\tau)\,\boldsymbol{N}_1(\tau)\,d\tau \ge 0.$$

Careful analysis of the partial differential equations involved lead to the result that $\mu_{jk}(0) = \text{const} \int_{\partial\Omega_1} \psi_{1,j}(\mathbf{y})\psi_{1,k}(\mathbf{y})\, dS(\mathbf{y})$, and from this, $\mu(0)$ is a positive definite matrix. By using a time discretization approach, these problems can be solved numerically [8].

3. GPU and multicore programming

The best computing performance in recent years in terms of Gflops per dollar have come from general purpose Graphical Processing Units (GPUs). GPUs provide an economical path to super computer performance, which today is measured in Teraflops (10^{12} flops) and Petaflops (10^{15} flops). In spite of the recent advances in support for developing software for GPUs (e.g., CUDA^{TM} and OpenCL^{TM}), the programming model for GPUs is somewhat limited, especially if the programmer aims for near optimal performance. While GPUs have support for hundreds of threads, features such as recursion are not available. Certain operations, such as parallel summation, must be programmed "by hand", taking into account the various levels of granularity of the GPU. For example, programming the optimal $O(\log n)$ parallel summation algorithm in CUDA requires separate kernel routines to handle within-thread-block summation, and another to handle within-grid summation.

As with any high-performance algorithm, it is important wherever possible to match the hardware data transfer rate to the rate at which the algorithm "consumes" data. This has the ironic result that many fast algorithms are hard to parallelize.

4. Implementation

The numerical methods developed in [8] for convolution complementarity problems lead to a sequence of complementarity problems [9, 10]:

$$0 \le \boldsymbol{N}_1^{(l)} \perp \boldsymbol{\varphi}_1^{(l)} + \sum_{j=0}^{l} \mu^{(l-j)} \boldsymbol{N}_1^{(j)} \ge 0, \tag{5}$$

for $l = 0, 1, \ldots$. The value $\mu^{(0)} \approx \mu(0)$, and is symmetric positive definite. This will form the main computational kernel of the method for simulating systems of elastic bodies. There are two parts to this computational kernel. One is computing the discrete convolution $\boldsymbol{\nu}_1^{(l)} := \sum_{j=0}^{l-1} \mu^{(l-j)} \boldsymbol{N}_1^{(j)}$. The second is to solve the complementarity problem

$$0 \le \boldsymbol{N}_1^{(l)} \perp \boldsymbol{\varphi}_1^{(l)} + \boldsymbol{\nu}_1^{(l)} + \mu^{(0)} \boldsymbol{N}_1^{(l)} \ge 0.$$

There are several ways to solve linear complementarity problems of this form. The oldest still commonly used is Lemke's method [11, 12], which is based on the simplex method for linear programming [13]. On the other hand, there are iterative methods [14, 15, 16] which can be separated into two different types. The older iterative methods for linear complementarity problems perform a nonlinear version of the Gauss–Seidel method with an iteration consisting of solving a sequence of simple scalar complementarity problems [14, 17]. A number of newer methods have been developed based on representing the linear complementarity problem in terms of (non-smooth) nonlinear functions, and then applying a version of Newton's method to the resulting system.

To explain more precisely how these methods work, consider the following linear complementarity problem: Find $\boldsymbol{z} \in \mathbb{R}^n$ satisfying

$$0 \le \boldsymbol{z} \perp M\boldsymbol{z} + \boldsymbol{q} \ge 0. \tag{6}$$

The older methods were similar to the following Gauss–Seidel method for :

```
for iter = 1, 2, ...
  for i = 1, 2, ..., n
    w_i ← Σ_{j=1; j≠i}^{n} m_ij z_j + q_i
    solve 0 ≤ z_i ⊥ m_ii z_i + w_i ≥ 0 for z_i
```

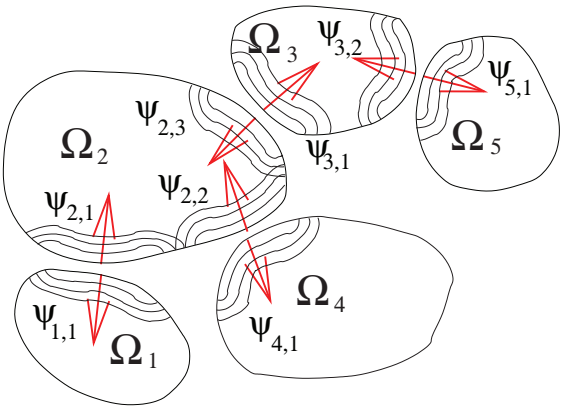

Figure 2: Interactions between contact regions and bodies

Solving a scalar linear complementarity problem $0 \le z \perp az + b \ge 0$ with $a > 0$ is easy: it is $z = \max(-b/a, 0)$. These methods converge rapidly if the matrix M is strongly diagonally dominant.

The newer methods instead represent the complementarity problem (6) as an equation, such as $\min(\boldsymbol{z}, M\boldsymbol{z}+\boldsymbol{q}) = 0$ where the minimum is taken componentwise. A version of Newton's method is used to solve this system of equations.

We use the older Gauss–Seidel based methods for solving the linear complementarity problems that arise in this model. The main reasons are that it is very straightforward to implement, and the matrix involved $\mu^{(0)} \approx \mu(0)$ is both sparse and strongly diagonally dominant for reasonable choices of the $\psi_{1,j}(\mathbf{x})$ functions. The interactions that appear in the linear complementarity problem for each step are illustrated in Figure 2 for the situation where the time step is a fraction of the time needed for elastic waves to traverse one body. The contour lines represent the elastic waves due to each patch $\psi_{i,j}$ (patch j on body i) and interactions can only occur between patches $\psi_{i,j}$ and $\psi_{p,q}$ if these contour lines of one patch (say $\psi_{i,j}$) overlap the other patch ($\psi_{p,q}$). Note that this is a symmetric relationship as $\mu(0)$ is a symmetric matrix. For contacting bodies, we have, for example, $N_{5,1} = N_{3,2}$ from Newton's second law. Thus in body Ω_2 there are direct interactions between patches $\psi_{2,1}$ and $\psi_{2,2}$, and between $\psi_{2,2}$ and $\psi_{2,3}$. However, there are no interactions between patches $\psi_{3,1}$ and $\psi_{3,2}$ in body Ω_3. In contrast, in rigid body dynamics, there are interactions between all the patches (and their forces) shown in Figure 2.

The matrix kernel function $\mu(t)$ for each body can be determined numerically via detailed finite element simulation. This simulation only needs to be done once for each patch, and can be done using a much smaller step size than the CCP computations. For symmetric bodies, such as balls, this can be further reduced as we only need to carry out one simulation for each class of symmetrically equivalent patches. (For example, we could use an icosahedral arrangement of patches on a ball, and we only need one finite element simulation to obtain the entire matrix kernel function $\mu(t)$.)

4.1. Implementation on GPUs

For GPU implementation, the convolution operation involves computing $\sum_{j=0}^{l-1} \mu_i^{(l-j)} \boldsymbol{N}_i^{(j)}$ where $\mu_i^{(k)} \approx \mu_i(hk)$, μ_i the matrix kernel function for body Ω_i. This involves matrix-vector multiplication together with parallel summation. The matrices involved are relatively small (for example, 20×20 using an icosahedral arrangement of patches on a ball), but this should significantly reduce the memory bandwidth needed between global device memory and shared memory.

Implementing the Gauss–Seidel iteration for the linear complementarity problem

$$0 \le \boldsymbol{N}_i^{(l)} \perp \boldsymbol{\varphi}_i^{(l)} + \boldsymbol{\nu}_i^{(l)} + \mu^{(0)} \boldsymbol{N}_i^{(l)} \ge 0$$

should exploit the sparsity of $\mu^{(0)}$. For example, if the bodies involved are balls and an icosahedral arrangement of the patches is used, $\mu^{(0)}$ should only have 6 non-zero entries per row out of 20 columns. It may be worthwhile hard-wiring (at least in some simulations) this sparsity pattern to avoid the many `if...then...` branches that would occur in navigating a general sparse matrix data structure.

5. Conclusions

GPUs offer a path to supercomputing performance without the expense. CCPs offer a path to simulating granular flows which keep the elastic behavior without the computational cost of finite element methods. The two together should provide an economical and effective way of accurately simulating large numbers of bodies that can impact and contact each other.

There are still a number of issues that need to be resolved. The first is that CCPs are based on a linearization which is valid for small displacements. Over time, bodies make large displacements. However, these displacements can be split into a rigid motion part and a truly elastic part:

$$\mathbf{u}_i(t,\mathbf{x}) = \overline{\mathbf{x}}_i(t) + Q_i(t)\left(\mathbf{x} - \overline{\mathbf{x}}_i(0)\right) + \mathbf{u}_i^{(e)}(t,\mathbf{x})$$

where $\overline{\mathbf{x}}_i(t)$ is the center of mass of body Ω_i at time t, $Q_i(t)$ is the orientation of the body at time t (represented as a 3×3 orthogonal matrix), and $\mathbf{u}_i^{(e)}(t,\mathbf{x})$ is the elastic part of the displacement. The other issue is frictional contact. This complicates the solution of the problem, especially as the matrix used in the linear complementarity problem is no longer symmetric. These are issues for future work.

References

[1] D. E. Stewart, Rigid-body dynamics with friction and impact, SIAM Review 42 (1) (2000) 3–39.
[2] W. J. Stronge, Impact Mechanics, 1st Edition, Cambridge Univesity Press, Cambridge, UK, 2000.
[3] D. Stoianovici, Y. Hurmuzlu, A critical study of the applicability of rigid-body collision theory, ASME J. Applied Mech. 63 (2) (1996) 307–316.
[4] A. Chatterjee, On the realism of complementarity conditions in rigid-body collisions, Nonlinear Dynamics 20 (2) (1999) 159–168.
[5] W. J. Stronge, Rigid body collisions with friction, Proc. Roy. Soc. London Ser. A 431 (1881) (1990) 169–181.
[6] E. J. Routh, A Treatise on the Dynamics of a System of Rigid Bodies, MacMillan, London, 1860.
[7] L. C. Evans, Partial differential equations, 2nd Edition, Vol. 19 of Graduate Studies in Mathematics, American Mathematical Society, Providence, RI, 2010.
[8] D. E. Stewart, Convolution complementarity problems with application to impact problems, IMA J. Applied Math. 71 (1) (2006) 92–119, first published Advance Access May 2005; DOI 10.1093/imamat/hxh087.
[9] R. W. Cottle, J.-S. Pang, R. E. Stone, The Linear Complementarity Problem, Academic Press, Boston, San Diego, New York, 1992, series on Computer Science and Scientific Computing.
[10] K. G. Murty, Linear and Combinatorial Programming, J. Wiley and Sons, 1976.
[11] C. G. Broyden, Lemke's method — a recursive approach, Lin. Alg. Appl. 136 (1990) 257–272.
[12] C. E. Lemke, J. J. T. Howson, Equilibrium points of bimatrix games, J. SIAM 12 (1964) 413–423.
[13] G. B. Dantzig, Linear Programming and Extensions, Princetn Uni. Press, Princeton, NJ, 1963.
[14] A. Brandt, C. W. Cryer, Multigrid algorithms for the solution of linear complementarity problems arising from free boundary problems, SIAM J. Sci. Statist. Comput. 4 (4) (1983) 655–684.
[15] A. Fischer, C. Kanzow, On finite termination of an iterative method for linear complementarity problems, Math. Programming 74 (3, Ser. A) (1996) 279–292.
[16] Y. Lin, C. W. Cryer, An alternating direction implicit algorithm for the solution of linear complementarity problems arising from free boundary problems, Appl. Math. Optim. 13 (1) (1985) 1–17.
[17] Y. Lin, C. W. Cryer, An alternating direction implicit algorithm for the solution of linear complementarity problems arising from free boundary problems, Appl. Math. Optim. 13 (1) (1985) 1–17.